

Методологія, теорія та практика соціологічного аналізу сучасного суспільства. Випуск 15

УДК 303. 645.063

КОМП'ЮТЕРНИЙ КОНТЕНТ-АНАЛІЗ: ПРОБЛЕМИ ТА ПЕРСПЕКТИВИ ВИРІШЕННЯ**Іванов Олег Валерійович** - аспірант кафедри соціології національного університету "Києво-Могилянська академія"

В статті розглядається методологічна цінність, переваги та недоліки контент-аналізу з використанням комп'ютерних програм. Метод розглядається в історичній перспективі; наводяться приклади сучасних контент-аналітичних досліджень із застосуванням комп'ютерних програм. Автор пропонує технічну концепцію контент-аналітичної програми — незалежної від мови аналізу, здатної виявляти значимі зв'язки між словами, визначати структуру аналізованих текстів.

В статье рассматривается методологическая ценность, преимущества и недостатки контент-анализа с применением компьютерных программ. Метод рассматривается в исторической перспективе; приводятся примеры современных контент-аналитических исследований с применением компьютерных программ. Автор предлагает техническую концепцию контент-аналитической программы — независимой от языка анализа, способной выявлять значимые связи между словами, определять структуру анализируемых текстов.

In the article methodological value, advantages and drawbacks of computer-aided content-analysis were shown. The method was reviewed in historical perspective; examples of modern computer-aided content-analytical studies were given. The author proposes technical conception of content-analytical software, which is language independent, capable of revealing meaningful connections between words and the structure of analyzed text.

Ключові слова: контент-аналіз, програмне забезпечення, автоматизація, кодування даних, аналіз текстів.

Інформація у вигляді тексту — досить поширена форма, в якій дослідник-соціолог отримує дані для аналізу. Це можуть бути відповіді на відкриті питання з анкет, транскрипти глибинних інтерв'ю чи фокус-груп, текстові масиви публікацій в ЗМІ та текстуалізовані відео- та аудіо записи різного походження. Текст — одна найбільш "насичених" форм вихідних даних, оскільки не має недоліків штучно створеної та формалізованої "реальності" опитувальників з обмеженим вибором відповідей. Водночас текстові дані — одні найбільш незручних для аналізу, оскільки потребують великих затрат часу, якщо йдеться про репрезентативні дослідження.

Один з найпоширеніших методів аналізу текстової інформації — контент-аналіз. Він дозволяє систематично та максимально об'єктивно досліджувати великі масиви документів (які можуть бути не лише текстами). Звичайно, ми не зводимо можливості контент-аналізу лише до кількісного опису очевидного змісту даних, оскільки грамотне його застосування може виявляти імпліцитну інформацію — мету автора, репрезентацію адресата повідомлення та ін. [1] Ми не переоцінюємо об'єктивність контент-аналізу, оскільки зміст¹ завжди є об'єктом інтерпретації, неупередженість якої обмежена в кожному разі. Втім, на сьогодні важко знайти кращий метод для ефективного дослідження тенденцій в великих масивах документів (найчастіше текстових чи вторинно вербалізованих).

Контент-аналіз не уникнув дилеми, притаманної нині будь-якій продуктивній діяльності людини: що краще — "ручна" робота чи автоматизована (з допомогою комп'ютерних програм)? Відповідь не така однозначна, як здається на перший погляд. При створенні "ручної" кодуючої схеми слід створити кодуючу книгу, де всі виміри категорій вичерпно пояснені та кодуючу форму, в яку будуть заноситись кодуючі результати. При побудові комп'ютерної кодуючої схеми також потрібна кодуюча книга (тобто файл) з вичерпним поясненням аналітичних словників та способу їх застосування. В деяких випадках можна застосовувати стандартні словники, які включені в програмні пакети, але сфера їх застосування — обмежена, крім того більшість з них — це словники англійські, в кращому випадку російськомовні, тому для аналізу, скажімо, українських текстів в будь-якому разі доводиться створювати свої словники.

¹ Англ. *content*, від чого й походить назва методу.

У випадку "ручного" кодування необхідний претест: кодування підвибірки матеріалів для того, щоб кодувальники потренувались застосовувати кодувальну схему, а також для того, щоб переконатись, що вони однаково розуміють виміри категорій. На цьому етапі варто переглянути кодувальну книгу та форму, якщо виникне така потреба. Із застосуванням комп'ютерних програм триангуляція не потрібна. На цьому етапі генеруються частотні ряди для кожної одиниці аналізу із використанням аналітичних словників. Виключається будь-яка ймовірність механічного, випадкового заповнення кодувальної форми.

Таким чином, застосування комп'ютерних програм дозволяє "пропустити" деякі процедурні етапи. Однак, аби при цьому не втратилась валідність та надійність отриманих даних, слід дуже чітко прописувати аналітичний словник, оскільки програма "розуміє" всі вхідні дані буквально і не буде "додумувати" пробіли, які міг допустити аналітик, складаючи кодувальну схему. Серед переваг систем автоматизованого кодування можна назвати:

1. Витрати на кодувальників дуже малі. Замість цілої групи можна використовувати одного.
2. Програма не має ні переконань, ні упереджень. Кодування відбувається за попередньо прописаною кодувальною схемою, без будь-якої реінтерпретації.
3. Комп'ютер може аналізувати величезні масиви даних, на кодування і аналіз яких людині потрібні були б місяці, а то й роки.

Серед недоліків можна назвати таке:

1. Комп'ютер кодує послідовність символів, заданих у аналітичному словнику, а не значення, яке вкладає в цю послідовність дослідник. З цього постає проблема семантичної валідності: чи може комп'ютер у відриві від контексту на основі послівної бази адекватно проаналізувати зміст тексту відповідно до поставлених дослідником задач?
2. Перед застосуванням аналітичного словника його слід перевірити на валідність, що все одно вимагає певних затрат на кодувальників.

На Заході ідея автоматизації систематичного аналізу текстів виникла досить давно. Гарвардська програма General Inquirer — перша широко застосовувана програма для автоматичного контент-аналізу — досі працює, переписана з оригінальної мови IBM PL/1 у середовищі Java [2].

1967 року у Філадельфії відбулась конференція під назвою "Анненберзька школа контент-аналізу, за підсумками якої були узгоджені ряд засадничих концептів та методичних рекомендацій щодо комп'ютерного контент-аналізу, на тодішньому рівні розвитку науки і техніки. З цього часу комп'ютерний контент-аналіз можна вважати інституціоналізованою в науковому середовищі методикою.

Після "методологічної ейфорії" 1960-х у 1970-ті та 1980-ті настав період деякої стагнації. Це було зумовлено тим, що спеціальних програм для мейнфреймів було дуже мало, документація для пересічних користувачів чи науковців без кібернетичної чи ІТ освіти була відсутня. Крім того, кожен мейнфрейм працював, фактично, лише з одним типом програм, а поняття "програмної сумісності" фактично не існувало. Крім того, тексти для аналізу мали бути подані у спеціальних форматах, при чому для кожної програми цей формат був свій, що перетворювало аналіз однієї й тієї ж інформації у безкінечне переписування. Тому той, хто хотів застосовувати контент-аналіз з використанням комп'ютерів мав перед тим стати професійним програмістом.

З введенням в широке користування персональних міні-комп'ютерів ситуація суттєво покращилась. З'явилися нові орієнтовані на кінцевого користувача програми. З поширенням мережі Інтернет у аналітиків з'явилось майже невичерпне джерело матеріалів для аналізу. Світова мережа стала поштовхом до революції в масштабності досліджень інформаційних потоків через доступність текстових даних, зазвичай стандартизованих у форматі HTML. Хоча такі джерела наразі не можуть бути проаналізовані напряму — тексти все ще мають бути експортовані з окремих сторінок Інтернет з використанням різного роду "веб-ріперів" (програм для виділення необхідної інформації із зовнішніх джерел) — однак програмні засоби для завантаження та фільтрування мережових потоків стають дедалі поширенішими, зручнішими в користуванні та універсальнішими. Як тільки така система буде налаштована на певні сайти, подальше здобуття необхідної інформації не обмежене в часі і вимагає затрат близьких до нуля.

Слід зазначити, однак, що найбільший розвиток автоматизованих інструментів дослідження саме в контексті соціальних наук відбувається в Європі, а не в США, хоча автоматична обробка природної людської мови ("*natural language processing*") — загальний перспективний напрямок світової інформаційної науки та технології [3, 4]. Сьогодні на Заході розвиваються експериментальні підходи до

аналізу: граматики залежностей (структура речень та їх зміст кодується залежно від зв'язків граматичного керування між словами) [5, 6], мережі конвенційного керування (поділ речень на іменникові та дієслівні конструції) [7], аналіз латентної семантики (використовує складний математичний апарат для виявлення груп документів, схожостей в потоці інформації) [8, 9, 10, 11].

Однак вказані вище підходи лише починають застосовуватись, а в переважній більшості контент-аналітичних досліджень із застосуванням програм використовуються системи трьох типів (наш умовний поділ) [12].

Перший тип — повністю автоматизовані пакети для контент-аналізу. Включають в себе розроблені авторами програм аналітичні словники, в які, зазвичай, неможливо або досить важко внести зміни. На нашу думку, такі програми можуть бути застосовані хіба що для перевірок лінгвістичних гіпотез, закладених в основу програмних словників та аналітичних схем. Адже сучасний рівень розвитку "штучного інтелекту" не дозволяє аналітику передавати програмі повний контроль над процесом інтерпретації масиву документів. Йдеться про те, що жодна сучасна програма не може *розуміти* текст в людському значенні цього поняття. Прикладами таких суцільно автоматичних "контент-аналітиків" є програми WordStat, Crawdad Text Analysis System, Diction, CATRAS та ВААЛ (одна з небагатьох програм для аналізу російськомовних та україномовних текстів).

Щодо останньої програми, як майже унікального випадку на пострадянському просторі, варто окремо висловити декілька зауважень. Програма є досить розрекламованою, видано монографію [13], присвячену її можливостям. Але чи справді ВААЛ є таким "всесильним" продуктом, який вартує 400 євро за 1 модуль (яких в програмі є 5) [14]? Автор концепції програми і одночасно автор відповідної монографії — Володимир Шалак — в своїй книзі не розглядає жодної контент-аналітичної системи, розробленої на заході чи на теренах колишнього СРСР, а лише ВААЛ. Таке ігнорування світового досвіду в сфері комп'ютерного аналізу текстів викликає великі сумніви у компетентності автора та розумінні ним реальних можливостей і обмежень методу.

Пошук в електронних базах даних статей на монографіях EBSCO, SocIndex та JSTOR показує, що програма не використовувалась жодним дослідником поза межами "трикутника" Україна-РФ-Білорусь, хоча мала б стати єдиним виходом для аналітиків, які працюють з неперекладеними англійською мовою джерелами. Пан Шалак не представив жодної публікації іноземною мовою з метою широкого обговорення поза межами достатньо замкнутої російськомовної наукової спільноти. Це викликає підозри, що автор просто не бажає обговорення свого "винаходу" спеціалістами світового рівня. Спеціалістів в галузі контент-аналізу на пострадянському просторі не так багато і жоден з них не представив критичного огляду програми.

Автор не наводить жодних обмежень застосування власної програми. Підхід, який не передбачає жодних обмежень, є догматичним і далеким від наукового критерію фальсифікабельності.

Методологія автора базується на припущенні, що текст може впливати на аудиторію самим своїм звучанням, тобто фоносемантичною структурою. Даний підхід є частиною ширшої теорії "нейролінгвістичного програмування", яка за визнанням світової наукової спільноти, є псевдонауковою. Нейролінгвістичне програмування іноді застосовується окремими психотерапевтами та політтехнологами, але жодних експериментальних чи інших валідних даних, які підтверджують припущення цієї теорії не існує.

Автор під час обґрунтування своєї методології посилається на доробок пані Анни Вежбицької, яка нині проживає в Австралії і про, нібито розроблену на основі її теорії, програму навряд чи знає. Програмним твором пані Вежбицької є праця "Семантичні примітиви"[15], в якій вона намагається винайти "мета-структуру" мови. Йдеться про пошук деяких семантичних універсалій для різних мов. Як визнає сама дослідниця пошук таких універсалій не є завершеним. Крім того, лінгвіст Вежбицька жодним чином не вважає, що "примітиви" можуть якимось чином впливати на сприйняття. Пан Шалак, на власний розсуд доповнивши "список Вежбицької", використовує "примітиви" як характеристики тексту щодо його впливу на аудиторію, що є абсолютно недопустимим актом "термінологічного жонглювання".

Програма "на виході" видає не числові параметри, які потребують подальшого аналізу дослідником, а готові характеристики тексту. Нам не відомі алгоритми, які в соціальних науках дозволяли би механічним їх застосуванням отримати готовий аналіз, який не потребує теоретичної інтерпретації. Таким чином, будь-яка програма у вигляді "чорного ящика" є радше "псевдонауковою комп'ютерною іграшкою", ніж серйозним інструментом аналізу.

На завершення аналізу концепції Володимира Шалака слід сказати, що представлені в книзі "прикладні застосування програми" є дослідженнями *ex post facto*, тобто надають широкі можливості "перебору" емпіричного матеріалу задля цільового пошуку підтвердження валідності висновків, які надає програма. Прогностична придатність програми перевіряється лише в одному випадку: аналізу висловлювань О. Лукашенка. Результатом аналізу є висновок: президент Білорусі "втомився від влади". Подальші дії президента в напрямку придушення опозиції якнайкраще підтверджують хибність проведеного "міні-дослідження".

Другий тип програм — надзвичайно прості та універсальні пакети підрахунку частот різних слів в текстах. До недоліків таких програм слід віднести відсутність категоризації підрахованих слів, що призводить до викривлення результатів в бік окремо беззмістовних слів, що належать до службових частин мови. Крім того, різні форми змістовних слів рахуються як різні слова, що не дозволяє поставити знак рівності між одиницями підрахунку та одиницями значень. Останній із згаданих недоліків був частково розв'язаний розробниками через процедуру лематизації (виділення незмінних частин слів). Однак подібних програм з лематичним вдосконаленням для слов'янських мов, які мають надзвичайно гнучку морфологію, яка передбачає досить відмінні форми для різних родів, чисел, відмінків і т.д., не існує. Прикладами другого типу програм є Yoshikoder, Concordance, HAMLET. Дані програми універсальні і не прив'язані до певної мови.

Третій тип програм — напівавтоматичні пакети для ручного кодування текстових даних. Вони спрощують деякі технічні аспекти кодування: користування ними аналогічне альтернативі "писати олівцем чи писати на комп'ютері". Дані комплекси позбавляють кодувальника постійно звертатись до кодувальної таблиці, упорядковують результати кодування багатьох кодувальників, зберігають дані в зручній для пошуку і маркування формі, візуалізують знайдені зв'язки. Звичайно, вони дозволяють заощадити певний час та підвищують ефективність, однак у випадку масивів інформації, що вимірюється сотнями і тисячами сторінок, дане програмне забезпечення відчутно не допоможе. Тому такі програми найчастіше використовуються для упорядкування якісного контент-аналізу та для контент-аналітичних кейс-стаді. Прикладами таких програм є Atlas.ti, MaxQDA та Transana (для мультимедійних документів).

Крім зазначених вище недоліків, жодна з вказаних програм не може аналізувати структуру тексту як цілісності та не може встановлювати однозначних зв'язків між словами в реченнях. Далі ми викладемо методичну пропозицію подолання цих обмежень через створення універсального пакету морфологічно-структурного аналізу. Ми пропонуємо об'єднати переваги другого і третього типу програм, але з певними вдосконаленнями.

Перш за все ми пропонуємо використовувати спеціальним чином структурований загальний лексичний словник замість наперед визначеного аналітичного словника. Такий підхід зробить програму універсальною для будь-якого дослідження і не "вбудовуватиме" в неї гіпотези розробників, які можуть відрізнятись, а то й вступати в суперечність з гіпотезами аналітиків. Словник має бути структурований у вигляді матриці, яка слугуватиме координатною сіткою, в межах якої програма працюватиме "читаючи" аналізований текст.

Матриця має концептуально виглядати як показано в таблиці 1. Строки відповідають всім словам, відмінним за значенням та морфологічною належністю (як і в будь-якому звичайному лексичному словнику). Колонка А містить числовий код кожного слова, який слугує внутрішнім посиланням на це слово в межах словника. Колонка В містить лему:

Таблиця 1

Матриця кодувального словника

A	B	C	D _{1..k}	D' _{1..j}	E	F	G	J	I
Індекс с слов а	Лема	Початкова форма	Форма відповідно го часу, роду, числа, відмінку, відміни і т.д.	Код граматичних характеристик	Код морфологічних характеристик	Синоніми	Антоніми	Контекстуальні, визначені користувачем синоніми	Контекстуальні, визначені користувачем антоніми
1... n		Лема+xx 0	Лема +xx						

даного слова, тобто незмінну, незалежно від граматичних форм, частину змістовного слова. Колонка С містить початкову форму слова, тобто ту форму, яку програма використовуватиме для видачі результатів підрахунку. Початкова форма утворюється додаванням до лемми деякої послідовності символів (літер) або без такого додавання ("нульове додавання").

Колонки $D_1 \dots D_k$ та $D'_1 \dots D'_j$ матриці містять пари слів та їх граматичних ідентифікаторів, таких як рід, число та інші граматичні характеристики слова. В даному разі $(1 \dots k)$ та $(1 \dots j)$ — повна парадигма кожної лексики. Кожна форма слова в межах словника комбінується за тим же принципом, що і початкова форма. Кожен ідентифікатор форми складається з унікальної для даної строки комбінації характеристик.

Для стандартизації внутрішніх словникових посилань кожен ідентифікатор форми може бути представлений як n -значне число, де n — кількість всіх можливих форм для всіх частин мови для даної мови. Таке число-код складається з цифр, починаючи з 1, які репрезентують значення кожної граматичної характеристики, та 0 — на позначення неможливості визначення даної характеристики для певної частини мови. В мовах з гнучким словотворенням слова матимуть достатньо велику кількість граматичних ідентифікаторів для їх чіткого визначення в межах матриці. Корисність такої можливості буде показана нижче.

Колонка E містить морфологічні ідентифікатори кожного слова, тобто його належність до певної частини мови. Тут ми використовуємо той самий принцип кодування, але з іншими значеннями. Це дозволить аналітику представляти результати частотного аналізу у вигляді категорій значимих слів. В будь-якій європейській мові іменники позначають деякі концепти чи феномени, прикметники представляють характеристики цих концептів чи феноменів, дієслова позначають дію певних іменників-суб'єктів і т.д. Таким чином на першому етапі обробки масиву текстових даних ми можемо з'ясувати, які концепти (іменники) є в текстах і яка їх частота.

В колонках F та J містяться посилання на лексичні синоніми та антоніми у формі лексичних індексів слів. Таким чином, "значеннєві кластери", тобто слова з усіма їх синонімами, можуть бути обраховані та узагальнені. Остання колонка I — визначені користувачем контекстуальні синоніми, які мають бути автоматично перекодовані у відповідні індекси слів. З метою виявлення таких контекстуальних синонімів можна поррахувати частоти значимих слів та логічно висувати, що деякі з них синонімами. В даному випадку, звичайно, не існує строгого правила щодо синонімізації. Втім, якщо, скажімо, ми маємо в одному тексті слово "президент" та його чи її ім'я, то ми можемо говорити, що слово "президент" і це ім'я є *контекстуальними* синонімами. Таким чином, варто спочатку зробити загальний частотний аналіз текстового масиву перед визначенням аналітичного словника, аніж теоретично, апіорно його конструювати.

Крім окремішньої функціональності кожної колонки, ми маємо користь зі спільного їх використання. Використовуючи порівняння індексів та позиційний аналіз на рівні речень ми можемо спробувати розв'язати одну з основних проблем автоматизованого контент-аналізу: "Чи справді певне слово прямо пов'язане з іншим словом, якщо вони зустрічаються в одному реченні?". Спосіб визначення прямого зв'язку передбачає два послідовних кроки: (1) ідентифікація позиції кожного значимого слова навколо ключового слова (слова, зв'язки якого з іншими словами перевіряються) в межах речення — це не важко зробити через фіксацію взаємного розташування пробілів та безперервних послідовностей літер з паралельною звіркою з матрицею словника; (2) порівняння морфологічних кодів ключового слова з відповідними кодами найближчих значимих слів. Правило визначення зв'язку таке: якщо ключове слово та одне з найближчих значимих слів мають однакові співмірні значення кодів і немає інших слів з тими ж кодами між ними, то ці слова пов'язані граматично і змістовно. Співмірність значень означає, що жодне значення, починаючи з 1, не може бути порівняне з нулем, який означає непридатність певної характеристики для даного слова.

Наразі контент-аналіз як метод обробки текстових даних використовується лише для аналізу виключно символічних послідовностей, тобто т.зв. "простого тексту" (plain text). Тому будь-який відповідним чином форматований текст переводиться в чисту послідовність символів перед проведенням аналізу. Втім, роблячи це ми втрачаємо значну частину інформації, яку містить формат частин тексту. Цю інформацію можна продуктивно використати для проведення структурного аналізу тексту.

Метою форматування тексту є виділення певних його частин через їх особливе значення чи важливість. Так в заголовках всі слова виділяються жирним шрифтом, кольором, великими літерами чи іншим способом; курсив використовується для наголошення на окремих словах, виділенні цитат і т.д. Виділенням всіх типів форматувань в тексті ми можемо визначити його змістовні та структурні елементи, а потім аналізувати кожен з елементів окремо чи аналізувати лише ті елементи тексту, які нас цікавлять. Так, ми можемо аналізувати лише заголовки, лише ті частини тексту, які мають заголовки з певними словами, лише цитати і т.д.

Запропонована нами концепція аналізу є лише пропозицією для технічного втілення. Поза сумнівом необхідно буде вирішити цілу низку проблем, перш ніж програмний пакет на основі цієї

концепції буде придатним для використання. Робота над цим ведеться в рамках розробки методології нашого дисертаційного дослідження та потребує конструктивної критики, порад, технічних пропозицій. Втім, відсутність значимих рішень висвітлених нами проблем в рамках вітчизняної дослідницької практики спонукає нас сподіватись на корисність потенційного результату.

ЛІТЕРАТУРА: 1. Детальніше обговорення характеристик контент-аналізу як методу див. Костенко, Наталія, Іванов, Валерій. Досвід контент-аналізу: Моделі та практики: Монографія. К.: Центр вільної преси, — 2003. Стр. 40-44. 2. Monroe, B.L., Schrodt, P. A. Introduction to the Special Issue: The Statistical Analysis of Political Text.// Political Analysis. — No. 16 — 2008. — Pp. 351–355. 3. Manning, C. D., Schutze, H. Foundations of Statistical Natural Language Processing. — MIT Press, 1999. 4. Bates, M. Models of natural language understanding.// Proceedings of the National Academy of Sciences of the United States of America. — Vol. 92, No. 22 — 1995. — Pp. 9977-9982. 5. Nivre, J. Sparse Data and Smoothing in Statistical Part-of-Speech Tagging.// Journal of Quantitative Linguistics. — No. 7(1) — 2000. — Pp. 1-17. 6. Nivre, J. Inductive Dependency Parsing. — Springer, 2006. 7. Chung-Hye H., Hedberg, N. Syntax and Semantics of It-Clefts: A Tree Adjoining Grammar Analysis.// Journal of Semantics. — No. 25(4) — 2008. — Pp. 345-380. 8. Landauer, T., Foltz, P. W., Laham, D. Introduction to Latent Semantic Analysis.// Discourse Processes — No. 25 — 1998. Pp. 259–284. 9. Deerwester, S., Dumais, S. Furnas G. W. et al. Indexing by Latent Semantic Analysis.// Journal of the American Society for Information Science. — No. 41 (6) — 1990. — Pp. 391–407. 10. T. Hofmann Probabilistic Latent Semantic Analysis. — 1999. — Режим доступу до статті: <http://lsirwww.epfl.ch/courses/dis/2003ws/papers/ut-cs-94-270.pdf>. 11. Berry, M., Dumais, S.T., O'Brien, G.W. (). Using Linear Algebra for Intelligent Information Retrieval. — 1995. — Режим доступу до статті: <http://lsirwww.epfl.ch/courses/dis/2003ws/papers/ut-cs-94-270.pdf>. 12. Alexa, M., Zuell, C. Text Analysis Software: Commonalities, Differences and Limitations: The Results of a Review.// Quality & Quantity — No. 34 — 2000. — Pp. 299-321. 13. Шалак В.І. Современный контент-анализ. Приложения в области: политологии, психологии, социологии, культурологии, экономики, рекламы. — М.: Омега - Л, 2009. — 272 с. 14. Проект ВААЛ. Прайс-лист [Електронний ресурс]. — Режим доступу: <http://www.vaal.ru/prog/price.php>. 15. Wierzbicka A. Semantic Primitives. — Frankfurt: Athenäum, 1972.

© О. В. Иванов, 2009